PR #25674 完整报告

sgl-project/sglang

[diffusion] Fix MOVA DAC bf16 on ROCm

合并时间：2026-05-22 15:18

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/25674

执行摘要

- 一句话：修复 ROCm bf16 下 DAC Snake 编译失败
- 推荐动作：该 PR 值得精读，特别是对于需要支持多硬件平台（如 ROCm）的团队。其设计模式——将 JIT 编译的函数拆分为纯 Python 实现和编译赋值，并添加条件回退——是一种优雅的跨平台兼容性解决方案，值得在其他类似场景中借鉴。

功能与动机

在 ROCm 平台上，DAC Snake 使用 torch.jit.script 编译 bf16 张量时，HIPRTC 编译失败。该问题在 PR #25411 的评论中被报告 (issuecomment-4476356483)。此 PR 旨在提供一个最小化修复，仅对 ROCm + bf16 场景回退到 *eager* 执行，避免 JIT 编译错误。

实现拆解

1. 将核心 snake 计算提取为纯 Python 函数 _snake：将原 @torch.jit.script 装饰的函数体移至一个普通的 Python 函数 _snake，该函数负责实际的 Snake 激活计算。
2. 保留 JIT 编译版本：通过 snake = torch.jit.script(_snake) 将 _snake 编译为 TorchScript，并作为默认的快速路径。对于非 ROCm bf16 场景，性能不受影响。
3. 新增条件判定函数 _should_use_eager_snake_on_rocm_bf16：该函数检查是否运行在 ROCm（torch.version.hip 非空）、张量是否在 GPU 上、以及数据是否为 bfloat16。
4. 在 Snake1d.forward 中引入分支：当条件满足时调用 _snake(x, self.alpha)（*eager* 模式），否则调用 snake(x, self.alpha)（JIT 加速版本）。
5. 文件变更：仅修改了 python/sglang/multimodal_gen/runtime/models/vaes/dac.py 一个文件，增加 16 行，删除 3 行，属于对已有行为的最小侵入性修复。

关键文件：

- python/sglang/multimodal_gen/runtime/models/vaes/dac.py（模块 VAE 模型；类别 source；类型 *core-logic*；符号 _snake, snake, _should_use_eager_snake_on_rocm_bf16）：唯一变更文件，核心修改包括拆分 snake 函数、新增条件判定逻辑、修改 Snake1d.forward 分支。

关键符号：_snake, _should_use_eager_snake_on_rocm_bf16

关键源码片段

python/sglang/multimodal_gen/runtime/models/vaes/dac.py

唯一变更文件，核心修改包括拆分 snake 函数、新增条件判定逻辑、修改 Snake1d.forward 分支。

```
from torch import nn

# 实现 Snake 激活函数的底层计算，以纯 Python 函数形式定义，
# 后续既可以用于 JIT 编译，也可以直接作为 eager 函数调用。
def _snake(x, alpha):
    shape = x.shape
    x = x.reshape(shape[0], shape[1], -1)
    x = x + (alpha + 1e-9).reciprocal() * torch.sin(alpha * x).pow(2)
    x = x.reshape(shape)
    return x

# Scripting this brings model speed up 1.4x
# 将 _snake 编译为 TorchScript，保留性能加速，作为默认路径。
snake = torch.jit.script(_snake)

# ROCm HIPRTC can fail to compile the scripted bf16 Snake kernel.
# 检测当前环境是否为 ROCm 且张量为 bf16，若是则触发 eager 回退。
def _should_use_eager_snake_on_rocm_bf16(x: torch.Tensor, alpha: torch.Tensor) -> bool:
    return (
        torch.version.hip is not None # 是否运行在 ROCm 平台
        and (x.is_cuda or alpha.is_cuda) # 张量是否在 GPU 上
        and (x.dtype == torch.bfloat16 or alpha.dtype == torch.bfloat16) # 是否为 bfloat16 类型
    )


class Snake1d(nn.Module):
    def __init__(self, channels):
        super().__init__()
        self.alpha = nn.Parameter(torch.ones(1, channels, 1))

    def forward(self, x):
        # 在 ROCm + bf16 条件下使用 eager 模式，避免 JIT 编译失败
        if _should_use_eager_snake_on_rocm_bf16(x, self.alpha):
            return _snake(x, self.alpha)
        # 其他路径走 JIT 编译加速版本
        return snake(x, self.alpha)
```

评论区精华

无争议性讨论。审核者 mickqian 直接批准了 PR，作者 qimcis 在 CI 通过后请求合并。没有 review 评论。

- 暂无高价值评论线程

风险与影响

- 风险：风险较低。修改仅影响 ROCm + bf16 场景下的 Snake 激活函数执行路径，对 NVIDIA 或非 bf16 路径无影响。JIT 路径完全保留，因此性能退化仅发生在受影响的场景。测试覆盖方面，没有新增单元测试，建议未来补充针对 ROCm bf16 的回归测试。
- 影响：影响范围有限，仅作用于 ROCm 平台上使用 DAC 模型且输入为 bf16 的用户。修复了此前导致推理失败或崩溃的编译错误，使 ROCm 用户能够正常使用 DAC 模型。对其他平台和数据类型无影响。
- 风险标记：缺少测试覆盖

关联脉络

- PR #25411 未知：此 PR 中引用了该 PR 的 issue comment，作为修复的动机来源。